

# UROP: A Simple, Near-Optimal Scheduling Policy for Energy Harvesting Sensors

Omer Melih Gul, and Elif Uysal-Biyikoglu, *Senior Member, IEEE*

## Abstract

This paper considers a single-hop wireless network where a central node (or fusion center, FC) collects data from a set of  $m$  energy harvesting (EH) nodes (e.g. nodes of a wireless sensor network). In each time slot,  $k$  of  $m$  nodes can be scheduled by the FC for transmission over  $k$  orthogonal channels. FC has no knowledge about EH processes and current battery states of nodes; however, it knows outcomes of previous transmission attempts. The objective is to find a low complexity scheduling policy that maximizes total throughput of the data backlogged system using the harvested energy, for all types (uniform, non-uniform, independent, correlated (i.e. Markovian), etc.) EH processes. Energy is assumed to be stored losslessly in the nodes batteries, up to a storage capacity (the infinite capacity case is also considered.) The problem is treated in finite and infinite problem horizons. A low-complexity policy, UROP (*Uniformizing Random Ordered Policy*) is proposed, whose near optimality is shown. Numerical examples indicate that under a reasonable-sized battery capacity, UROP uses the arriving energy with almost perfect efficiency. As the problem is a restless multi-armed bandit (RMAB) problem with an average reward criterion, UROP may have a wider application area than communication networks.

## Index Terms

communication networks, decision theory, energy harvesting, scheduling algorithms, wireless sensor network

O. Melih Gul and Elif Uysal-Biyikoglu are with the Department of Electrical and Electronics Engineering, Middle East Technical University, Ankara, Turkey (omgul@metu.edu.tr, uelif@metu.edu.tr). A preliminary version of this paper has been accepted to appear at the 2014 IEEE Wireless Communications and Networking Conference (WCNC 2014), Istanbul, Turkey.

## I. INTRODUCTION

POWER resource and battery lifetime are important issues for networks such as Wireless Sensor Networks (WSNs). Energy harvesting (EH) [1] can enable WSN operation in environments where maintenance is impractical or too costly. Energy harvesting (EH) extends reliable operation lifetime [2], [3]. Energy may be harvested from the environment in many different ways (solar, kinetic, etc.) [4]. Since energy harvesters generally depend on uncontrollable energy resources and the amount of harvested energy is generally low [4], WSNs need robust, environmentally adaptive, energy efficient policies for their operations.

In this paper, we consider a WSN where a fusion center (FC) collects data from  $m$  EH sensor nodes by assigning the nodes to  $k$  orthogonal communication channels in each time slot. It is assumed that each node always has data to transmit (i.e., nodes are data backlogged). Each node has a battery (of a certain capacity, and without leakage) to store harvested energy. It is also assumed that the multi-access communication is error-free and there is no fading. If a node is scheduled, it will be assigned one of the channels. When a node is scheduled to transmit, it can transmit data to the FC if it has sufficient energy to send a packet. The transmission of each packet lasts an entire time slot. The objective of the FC is to maximize the total throughput over a finite or infinite problem horizon.

In practice, battery states of nodes could be made available to the FC through some additional cost (i.e. feedback) and complexity. However, it is interesting from a practical perspective to consider the case where the FC makes scheduling decisions without knowledge of the instantaneous battery states at nodes, or their statistics. Fortunately, it turns out that this lack of knowledge has little effect on performance. We will observe that by knowing only the outcomes of previous transmission attempts, the FC can schedule almost as efficiently as an omniscient scheduler.

This problem may be formulated as a partially observable Markov Decision Process (POMDP), and dynamic Programming (DP) [5] can be employed for optimal solution. However, DP has exponential complexity with respect to number of nodes  $m$  [5]. Furthermore, the state space of DP should be very large to get a good approximation to the problems with continuous state variables like energy. Therefore, complexity of DP becomes excessively high for the EH scheduling problem with large number of nodes.

A second approach for solving this scheduling problem is reinforcement learning by con-

sidering the problem as a POMDP. Q-learning [6] is the easiest to implement and the most effective model-free algorithm among reinforcement learning algorithms. Q-learning guarantees convergence to optimal for a generic model. However, Q-learning is not applicable for problems with large-state space because its convergence is slow [7]. In fact, many algorithms can guarantee the convergence to optimal behavior [8]. However, in many practical applications, a policy which achieves near optimality quickly is preferable to the policy which converges slowly to exact optimality [7]. As the discount factor gets closer to 1 (i.e. the undiscounted case), the convergence rate of Q-learning decreases more. There are approaches such as R-learning [9] which maximize average reward; however, the convergence of R-learning has not been proven. Also, reinforcement learning has a very important problem: *the trade-off between exploration and exploitation* [10]. Therefore, Q-learning and generally reinforcement learning do not seem to be suitable for obtaining an efficient and practical solution to this scheduling problem, especially a large number of sensors and a continuous state variable, energy, is considered.

Another approach for this scheduling problem is to consider it as a restless multi-armed bandit problem (RMAB) which is a special version of POMDP. RMAB is an extension to classical multi-armed bandit problem which is solved optimally by Gittins [11] and an optimal solution is proposed under certain assumptions by Whittle [12]. Papadimitriou and Tsitsiklis show that finding optimal solution to a general RMAB is PSPACE-hard and it has a very high computational complexity [13]. Considering memory limits of sensors, a much more applicable policy is required. Therefore, a simpler approach called a myopic policy (MP) is suggested for RMAB problems and proven to be optimal in limited cases for the sensor management problems in [14], [15], [16]. However, a myopic policy is not generally optimal since MP concentrates only on the present and not consider the future [17], [18]. A channel probing problem is studied in [19] and it is shown that MP is not always optimal. The assumption that the scheduling decision does not affect transition probabilities was an appropriate one for the problems addressed in [14], [15], [16], [19]. However, for the EH scheduling problem at hand, this is not a reasonable assumption, as energy is a flexible resource that can be stored without any discount (ignoring battery leakage which is very minor in practice [2]) and can be used whenever desired. Therefore, the solutions presented in [14], [15], [16], [19] papers are not directly applicable to our problem.

The closest works in the literature to the problem at hand are the scheduling problems studied in [18], [20]. We have posed essentially the same problem, with the exception that no battery and

unit sized batteries at nodes are assumed in [18] and [20], respectively. In both [18] and [20], the scheduling problem is formulated as a POMDP where the focus is on immediate reward instead of future rewards. In [20], a single-hop wireless sensor network which consists of EH transmitter nodes with a unit sized battery and a central receiver node with multi server is considered as a restless multi-armed bandit problem (RMAB). Optimality of Whittle index policy which is generally suboptimal for RMAB [21] is proven for a certain case under certain assumptions on the EH process. In [20], the optimality of a Round-Robin based myopic policy is proved under the assumption that each node has only unit sized battery and the ratio between the number of transmitter nodes and the number of communication channels of the central node is an integer ( $m/k$  is an integer). In [18], the problem is formulated as POMDP and the optimality of MP is proven for two cases: 1) the nodes are not able to harvest and transmit simultaneously, and the EH process transition probabilities are affected by the scheduling decisions, and 2) the nodes have no battery. Since myopic policies proposed in [18] and [20] are based on Round-Robin(RR) Scheme, assuming that  $p = m/k$  is an integer is important (also period of RR Policy). These assumptions are somewhat restrictive for real life implementation.

To set up the problem, a model about the generation and usage of energy is needed. First, energy in a node's battery decreases if the node sends a data packet. Second, energy in a battery increases in a continuous fashion by harvested energy. Third, battery leakage is neglected. This assumption follows from examining typical batteries in use today for which leakage is negligibly small for over durations of several minutes. Based on these mild assumptions about energy, a suitable performance measure for a policy can be average reward over the finite and the infinite horizon rather than expected total discounted reward for this scheduling problem which is a delay-insensitive communication problem [22]. In communication network problems, delay issue is investigated as average delay not as discount. In applications, EH sources may use vibrational or kinetic energy, the behavior of which is typically not predictable [1], [3]. Optimal scheduling for this continuous, independent EH process becomes a hard problem, and the problem requires good near-optimal solutions.

By taking a deterministic approach, a near-optimal transmission scheduling policy, Uniforming Random Ordered Policy (UROP), is developed by assuming that each sensor has an infinite capacity battery (It will be shown that if the sensors have a reasonable-sized finite battery, UROP has almost same efficiency as its efficiency under a reasonable-sized finite battery assumption).

It is also guaranteed that UROP is *asymptotically optimal* for a general case of energy arrival process under the infinite battery assumption (larger than unit battery) as the horizon length increases. In comparison with the myopic policies in [18] and [20], UROP can still guarantee near-optimal performance when  $p = m/k$  is not an integer.

The rest of this paper is organized as follows. The system model and problem formulation are described in Section II. In Section III, we study the scheduling capacity. In section IV, we show that Round-Robin based policies cannot guarantee 100% throughput under many non-uniform energy harvesting process for nodes. We show the optimal omniscient solution for this problem in section V. In section VI, we suggest a novel, low-complexity scheduling policy which is nearly throughput optimal for quite general EH processes (uniform, non-uniform, independent, correlated) in a finite horizon problem under an infinite battery assumption. Next, efficiency bounds on UROP are obtained. Section VII extends the results from finite horizon to infinite horizon. In section VIII, we compare the performance of UROP with that of a Round-Robin policy and the Myopic Policy in [18], [20] through simulations. Section IX concludes the paper.

## II. SYSTEM MODEL AND PROBLEM FORMULATION

We consider a single-hop wireless network in which  $m$  energy harvesting (EH)-capable sensors have circularly symmetric distribution around a Fusion Center (FC) and send data packets to FC (see Figure 1). The WSN operates in a time-slotted fashion over time slots (TSs) of equal duration. In each TS, FC schedules  $k$  of  $m$  sensors for data transmission by assigning each one of  $k$  orthogonal channels. We assume that each sensor always has data to transmit (i.e. data is backlogged as in [18] and [20]) during the problem horizon of  $N$  TSs. Data packets have equal size and require *unit energy* for transmission.

The EH processes are assumed to be independent for each node. The total energy harvested by sensor  $s_i$  by time  $t$  is denoted by  $E_i^{tot}(t)$ , and the increment of this energy harvested during the TS  $t$  is denoted as  $E_i^h(t)$ . The energy present in the battery at  $t$  (stored minus used) is  $B_i(t)$ .

We denote by  $S_m = \{s_1, s_2, \dots, s_m\}$  and  $A_k = \{a_1, a_2, \dots, a_k\}$ , the set of all sensors and the set of orthogonal channels, respectively. The amount of data sent by node  $s_i$  in TS  $t$  is denoted by  $D_i(t) = 1(s_i \in S(t))1(B_i(t) \geq 1) \in \{0, 1\}$  where  $1(A)$  is the indicator function of event  $A$ , and  $S(t) \subset S_m$  is the set of  $k$  nodes scheduled at  $t$ . The set  $S(t)$  is determined by a policy  $\pi$ .

Two definitions are in order: A *fully efficient policy* (alternatively, a 100% *efficient policy*)

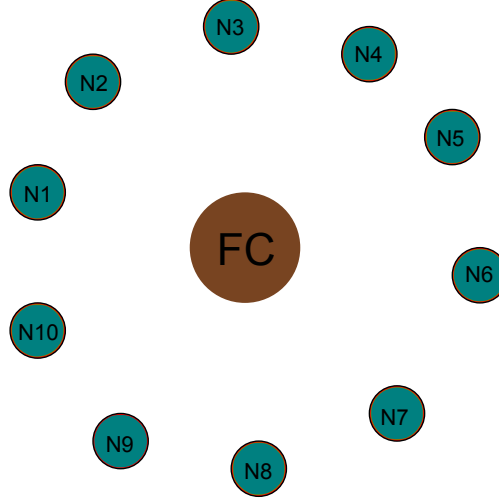


Fig. 1. A single hop wireless sensor network where a fusion center (FC) collects data from energy harvesting (EH) nodes located in a star topology around it.

ensures that the nodes use up all of the harvested energy by the end of the problem horizon, more precisely,  $B_i(N) < 1$  for  $\forall s_i \in S_m$ . An *optimal policy* is one that maximizes data throughput for the given sequence of energy harvests. For certain energy harvest processes, an optimal policy may not be fully efficient, as we will be clear in the next section.

Consistently with previous literature, the general objective is to maximize the expected discounted reward over the problem horizon:

$$\max_{S(t), t=1, \dots, N} V_{\text{tot}}(t) = \max_{s(t), t=1, \dots, N} E \left[ \sum_{t=1}^N \beta^{t-1} \sum_{s_i \in S_m} 1(s_i \in S(t)) 1(B_i(t) \geq 1) \right] \quad (1)$$

where  $0 < \beta \leq 1$  is the discount factor, which reduces the value of data sent later. The discount factor corresponds to placing lower value on data that is delayed. However, note that the problem at hand assumes infinite backlog and is therefore delay insensitive by nature. The discount could also be considered to model battery leakage that happens as transmission is withheld. Therefore, average reward criterion is more suitable measure for delay-insensitive communication problems like this scheduling problem than discounted reward criterion [22]. Consistently with our assumptions about infinite data buffers, and no battery leakage, we shall set  $\beta = 1$  and convert the objective function in (1) to that in (2), which is an average reward criterion.

$$\max_{S(t), t=1, \dots, N} \frac{V_{\text{tot}}(t)}{N} = \max_{s(t), t=1, \dots, N} E \left[ \frac{1}{N} \sum_{t=1}^N \sum_{s_i \in S_m} 1(s_i \in S(t)) 1(B_i(t) \geq 1) \right] \quad (2)$$

We propose an algorithm, UROP, which achieves nearly 100% throughput (and 100% efficiency whenever a fully efficient schedule is feasible) in a broad class of energy harvesting (arrival) process under infinite battery assumption. In this work, efficiency of a policy  $\pi^*$  ( $\eta_*$ ) is defined as ratio of total throughput by the policy  $\pi^*$  to total throughput by fully efficient policy  $\pi^{fe}$  on the problem horizon ( $\eta_{fe} = 1$ ). In section III, it is proven that efficiency of an arbitrary Round Robin Policy with quantum=1  $\pi^{RR}$  is very close to that of myopic policy  $\pi^{MP}$  ( $\eta_{MP}$ ) proposed in [18] and [20]. Therefore, the efficiency of UROP  $\pi^{UROP}$  ( $\eta_{UROP}$ ) will be compared with that of an arbitrary Round Robin Policy with quantum=1  $\pi^{RR1}$  ( $\eta_{RR}$ ) in section VII for simplicity.

An arrival process is called *admissible* if a fully efficient schedule is possible. By the analogy with admissible processes in these problems, we introduce four new terms which we use for the EH scheduling problem in the rest of paper. *Density of sensor  $i$* ,  $D_i$ , is the total number of packets sent by the sensor  $s_i$  with  $\pi^{fe}$  normalized by  $\frac{kN}{m}$  during problem horizon  $N$ . *Partial Density of sensor  $s_i$* ,  $D_i^{(T)}$ , is the total number of packets sent by the sensor  $s_i$  with  $\pi^{fe}$  normalized by  $\frac{k(N-T)}{m}$  in the interval  $[T, N]$ . *Density ( $D$ )* is the average of densities of all sensors during problem horizon  $N$  (see Equation (3)). *Partial Density ( $D^{(T)}$ )* is the average of partial densities of all sensors in the interval  $[T, N]$  (see Equation (4)). By definition,  $D, D^{(T)} \leq 1$ .

$$D = \frac{\sum_{s_i \in S_m} D_i}{m} \quad (3)$$

$$D^{(T)} = \frac{\sum_{s_i \in S_m} D_i^{(T)}}{m} \quad (4)$$

### III. SCHEDULING CAPACITY

To find a robust, efficient scheduling policy, we need to consider scheduling capacity of the FC. Scheduling capacity corresponds to the maximum number of nodes which can be scheduled by the FC in one TS. Since FC has  $k$  orthogonal channels, the scheduling capacity of the FC is  $k$ . If the amount of average harvested energy is so high that the scheduling capacity is exceeded, no 100% efficient policy exists and energy will keep accumulating (there is an energy surplus). Considering finite batteries, this will cause overflow in the batteries of nodes. Theorem 1 explores the region of energy harvest rates such that a 100% efficient policy is feasible.

We shall now make some definitions that will be used in the rest of this section and the paper. We denote by  $V_i^{(T)}(t)$  and  $V_{av}^{(T)}(t)$ , the energy available to sensor  $s_i$  in the interval  $(T, N]$  and



the average of this over all sensors in this time interval, respectively. Also, we denote by  $V_{tot}^{(T)}(t)$  the total energy available  $(T, N]$ :

$$V_i^{(T)}(t) = \left\lfloor B_i(T) + \sum_{t=T+1}^N E_i^h(t) \right\rfloor \quad (5)$$

$$V_{tot}^{(T)}(t) = \sum_{s_i \in S_m} V_i^{(T)} \quad (6)$$

$$V_{av}^{(T)}(t) = \frac{\sum_{s_i \in S_m} V_i^{(T)}}{m} \quad (7)$$

**Theorem 1 (Scheduling Capacity Theorem):** For  $0 \leq T < N$ ,

(i) If  $V_{av}^{(T)}(t) > \frac{k(N-T)}{m}$ , all possible policies will have efficiency below 100% and battery levels of some sensors grow unboundedly (in practice, considering finite batteries, they will overflow.)

(ii) If  $V_{av}^{(T)}(t) \leq \frac{k(N-T)}{m}$ , a 100% efficient policy that maximizes throughput while keeping battery levels of all sensors finite, exists.

**Proof of Theorem 1:**

(i) In this case,

$$V_{tot}^{(T)}(t) = mV_{av}^{(T)}(t) > k(N-T) \quad (8)$$

As the total uplink rate available is  $k$  data packets per slot, FC can accumulate at most  $k(N-T)$  packets from the nodes in the interval  $(T, N]$ . Suppose that there is a policy  $\pi^*$  which can achieve up to scheduling capacity. Then, efficiency of  $\pi^*$  equals to the maximum efficiency in the conditions (8), and it is represented as below:

$$\eta_{max} = \eta_* = \frac{\min \{k(N-T), V_{tot}^{(T)}(t)\}}{V_{tot}^{(T)}(t)} = \frac{k(N-T)}{V_{tot}^{(T)}(t)} \quad (9)$$

If (8) is satisfied, the scheduling capacity is exceeded in the interval  $(T, N]$ . By (8),  $\eta_{max} = \eta_* < 1$ . Hence, there is no 100% efficient policy which lets FC receive  $V_{tot}^{(T)}(t)$  packets from the nodes. By definition of  $D^{(T)}$ , (8) is equivalent to  $D^{(T)} > 1$ . Define excess energy as  $B_{ex}(t) = \sum_{i=1}^m \lfloor B_i(t) \rfloor = \min \{0, V_{tot}^{(T)}(t) - k(N-T)\}$ . By definition of  $D^{(T)}$ ,  $B_{ex}(t) = k(N-T)(D^{(T)} - 1)$ .  $D^{(T)} > 1$ , and  $N \rightarrow \infty$ , battery levels of some sensors grow unboundedly.

(ii) In this case,

$$V_{tot}^{(T)}(t) = mV_{av}^{(T)}(t) \leq k(N-T) \quad (10)$$



FC can receive maximum  $k(N - T)$  data packets from the nodes in the interval  $(T, N]$ . An omniscient policy could fill up all channels in all time slots as long as there is a sensor with available energy. Trivially, this achieves 100% efficiency if  $D^{(T)} \leq 1$  (equivalent to 10). It is summarized below:

$$\eta_{max} = \eta_* = \frac{\min \left\{ k(N - T), V_{tot}^{(T)}(t) \right\}}{V_{tot}^{(T)}(t)} = \frac{V_{tot}^{(T)}(t)}{V_{tot}^{(T)}(t)} = 1 \quad (11)$$

By (11),  $\eta_{max} = \eta_* = 1$ . Hence, there is a fully efficient (100% efficient) policy which makes FC receive  $V_{tot}^{(T)}(t)$  packets. Battery levels of all sensors are kept finite. Hence when  $D^{(T)} \leq 1$ , there is an optimal policy which is 100% efficient.

#### IV. EFFICIENCY OF RR-BASED POLICIES

The scheduling problem in this paper are also studied in [18] and [20] for certain specific cases. Both papers propose RR-based policies with quantum=1 which are myopic policies. Then, they prove the optimality of these policies under certain specific cases.

First, we will investigate the efficiency of RR-based policies by Theorem 2. Then, we will prove by Theorem 3 that there is only a slight difference between the efficiencies of any two RR-based policies in long problem horizon  $N(\frac{m}{k} = p \ll N)$ . Hence, the efficiency of RR-based myopic policies in [18] and [20] are investigated. It is shown that the policies in [18] and [20] are generally suboptimal.

For the cases that each node has a battery larger than unit size, there is no known myopic policy in the literature. Therefore, we will compare UROP only with the policies in [18] and [20], and the optimal policy in this paper.

**Theorem 2:** *Suppose that  $N \gg p = \frac{m}{k} \in \mathbb{Z}$  and  $V_{av}^{(T)}(t) \leq \frac{k(N-T)}{m}$ . If there are some sensors  $s_i \in S_m$  such that  $V_i^{(T)}(t) > \frac{k(N-T)}{m}$ , all RR-based policies with quantum=1 will have efficiency below 100% although a fully efficient policy ( $\pi^{fe}$ ) exists. Moreover, batteries of some sensors will overflow.*

**Proof of Theorem 2:** In this proof, what is implied by RR policy is RR-based policies with quantum=1. We investigate efficiency of RR in the two possible cases:

- i. If  $\sigma = \frac{kN}{m} = \frac{N}{p} \in \mathbb{Z}$ , RR allocates each node  $\sigma$  TSs for transmission.
- ii. If  $\sigma = \frac{kN}{m} = \frac{N}{p} \notin \mathbb{Z}$ , RR allocates some nodes  $\lfloor \sigma \rfloor + 1$  TSs and other nodes  $\lfloor \sigma \rfloor$  TSs.

Assume that there are some nodes  $s_i \in S_m$  such that  $V_i^{(T)}(t) > \frac{k(N-T)}{m}$ . We denote by  $H$  the set of these sensors. By definition  $D_i^{(T)} > 1$  for nodes  $s_i \in H$ .

**Case i:** If the FC schedules  $m$  sensors by RR policy in the problem horizon  $N$ , RR policy allocates each node  $\sigma = \frac{kN}{m} = \frac{N}{p}$  TSs equally. Although  $V_{av}^{(T)}(t) \leq \frac{k(N-T)}{m}$ , each sensor  $s_i \in H$  can transmit maximum  $\sigma$  data but cannot transmit  $V_i^{(T)}(t) - \sigma$  data due to RR policy. On the other hand, each of other sensors  $s_i \in S_m - H$  can transmit all  $V_i^{(T)}(t)$  packets. By analogy with scheduling capacity, the efficiency of a RR policy can be represented as below:

$$\eta_{RR} = \frac{\sum_{s_i \in S_m} \min \{V_i^{(T)}(t), \sigma\}}{\sum_{s_i \in S_m} V_i^{(T)}(t)} \quad (12)$$

$$= \frac{\sum_{s_i \in H} \min \{V_i^{(T)}(t), \sigma\} + \sum_{s_i \in S_m - H} \min \{V_i^{(T)}(t), \sigma\}}{\sum_{s_i \in S_m} V_i^{(T)}(t)} \quad (13)$$

$$= \frac{\sum_{s_i \in H} \sigma + \sum_{s_i \in S_m - H} V_i^{(T)}(t)}{\sum_{s_i \in S_m} V_i^{(T)}(t)} \quad (14)$$

$$= 1 - \frac{\sum_{s_i \in H} (V_i^{(T)}(t) - \sigma)}{\sum_{s_i \in S_m} V_i^{(T)}(t)} \quad (15)$$

Considering the definition of  $D_i^{(T)}$ ,  $\eta_{RR}$  can also be represented as below:

$$\eta_{RR} = 1 - \frac{\sum_{s_i \in H} (D_i^{(T)} - 1)}{\sum_{s_i \in S_m} D_i^{(T)}} \quad (16)$$

Since  $D_i^{(T)} > 1$  for  $s_i \in H$ ,  $\eta_{RR} < 1$ . Hence, suboptimality of RR policy is proven for the first case although there exists an 100% efficient policy by Theorem 1.

**Case ii:** If the FC schedules  $m$  sensors by RR policy in the problem horizon  $N$ , RR policy allocates some nodes  $\lfloor \sigma \rfloor + 1$  TSs and other nodes  $\lfloor \sigma \rfloor$  TSs for transmission where  $\sigma = \frac{kN}{m} = \frac{N}{p} \notin \mathbb{Z}$  and  $\sigma^+ = \sigma - \lfloor \sigma \rfloor$ . To maximize efficiency of RR policy, we assume that each sensor  $s_i \in H$  can transmit maximum  $\lfloor \sigma \rfloor + 1$  data. However, each of these sensors cannot transmit  $V_i^{(T)}(t) - \lfloor \sigma \rfloor - 1$  data due to RR policy although  $V_{av}^{(T)}(t) \leq \frac{k(N-T)}{m}$ . On the other hand, each of other sensors  $s_i \in S_m - H$  can transmit all  $V_i^{(T)}(t)$  data. By the analogy with scheduling capacity, the efficiency of RR policy can be represented as below:

$$\eta_{RR} = \frac{\sum_{s_i \in S_m} \min \{V_i^{(T)}(t), \sigma\}}{\sum_{s_i \in S_m} V_i^{(T)}(t)} \quad (17)$$

$$= \frac{\sum_{s_i \in H} \min \{V_i^{(T)}(t), \sigma\} + \sum_{s_i \in S_m - H} \min \{V_i^{(T)}(t), \sigma\}}{\sum_{s_i \in S_m} V_i^{(T)}(t)} \quad (18)$$

$$= \frac{\sum_{s_i \in H} (\lfloor \sigma \rfloor + 1) + \sum_{s_i \in S_m - H} V_i^{(T)}(t)}{\sum_{s_i \in S_m} V_i^{(T)}(t)} \quad (19)$$

$$= 1 - \frac{\sum_{s_i \in H} (V_i^{(T)}(t) - \lfloor \sigma \rfloor - 1)}{\sum_{s_i \in S_m} V_i^{(T)}(t)} \quad (20)$$

Considering the definition of  $D_i^{(T)}$ ,  $\eta_{RR}$  can also be represented as below:

$$\eta_{RR} = 1 - \frac{\sum_{s_i \in H} (D_i^{(T)} \sigma - \lfloor \sigma \rfloor - 1)}{\sum_{s_i \in S_m} D_i^{(T)} \sigma} \quad (21)$$

$$= 1 - \frac{\sum_{s_i \in H} (D_i^{(T)} - 1) \sigma - (1 - \sigma^+)}{\sum_{s_i \in S_m} D_i^{(T)} \sigma} \quad (22)$$

It is known that  $D_i^{(T)} > 1$  for  $s_i \in H$  and  $(1 - \sigma^+) < (D_i^{(T)} - 1) \sigma$  since  $\sigma \gg 1 > 1 - \sigma^+$ . Therefore,  $\eta_{RR} < 1$ . Hence, suboptimality of RR policy is also proven for the second case although there exists an 100% efficient policy by Theorem 1. From (16), (22) and Theorem 1, efficiency of RR be as low as  $\frac{k}{m}$ . This worst case efficiency of  $\frac{k}{m}$  occurs when  $k$  of the nodes always have sufficient energy to transmit a packet in each TS and the remaining ones have no energy.

For a sufficiently long problem horizons, these results can be extended to RR-based policies with larger quanta. The following remark, used in the rest of the paper, is a consequence of the assumption there is no battery leakage.

**Remark 1 (No battery leakage):** Let  $T_1, T_2 \in (0, N]$  and  $T_1 < T_2$ . If  $s_i$  is not scheduled (selected) in interval  $(T_1, T_2]$ ,  $B_i(T_1) \leq B_i(T_2)$  where  $B_i(t)$  is the energy remaining in battery of sensor  $s_i$  at the end of TS  $t$ . That is,  $B_i(t)$  does not decrease unless  $s_i$  transmits data.

Theorem 2 states that RR-based policies become suboptimal when  $D_i^{(T)} > 1$  even for one node. The myopic policies (MP) in [18] and [20] are RR policies with quantum=1. In Theorem 3, it is shown that the MPs have almost same efficiency as any other RR policy with quantum=1.

**Theorem 3 (Upper and lower bounds on RR throughput ):** Assume that  $\frac{m}{k} \in \mathbb{Z}$ . In problem horizon  $N$ ,

$$\max \{V_{tot}^{RR}(N)\} - \min \{V_{tot}^{RR}(N)\} \leq m - k \quad (23)$$

where  $\min \{V_{tot}^{RR}(N)\}$  and  $\max \{V_{tot}^{RR}(N)\}$  are the minimum and maximum throughput which can be achieved under a RR policy with quantum=1, respectively.

**Proof of Theorem 3:** There are three cases for the problem horizon  $N$ : 1)  $N < \frac{m}{k}$ , 2)  $N \geq \frac{m}{k}$  and  $\frac{kN}{m} \in \mathbb{Z}$ , and 3)  $N \geq \frac{m}{k}$  and  $\frac{kN}{m} \notin \mathbb{Z}$

**Case 1:** If  $N < \frac{m}{k} = p$ ,  $N \leq p - 1$ . Since  $\min \{V_{tot}^{RR}(N)\} \geq 0$  and  $\max \{V_{tot}^{RR}(N)\} \leq kN \leq k(p - 1) = m - k$ ,  $\max \{V_{tot}^{RR}(N)\} - \min \{V_{tot}^{RR}(N)\} \leq m - k$ . This proves the statement for this case.

**Case 2:** Denote by  $U_i$  the nodes scheduled in TS  $i$  where  $i \leq p$  and  $S_m = \bigcup_{i=1}^p U_i$ . All RR policies have same length period  $p$ . Denoted by  $\tau_l^{RR}$  the  $l^{th}$  period of RR, namely,  $\tau_l^{RR} = [(l - 1)p + 1, lp]$ . Assume that  $T_1, T_2 \in \tau_j^{RR}$  and  $T_1 < T_2$  where  $T_1$  and  $T_2$  are the TSs when a node  $s_j$  is scheduled  $l^{th}$  time by the FC under two different myopic policies,  $\pi^{RR1}$  and  $\pi^{RR2}$ , respectively.

By Remark 1, efficiency of  $\pi^{RR2}$  in  $T_2$  is not lower than that of  $\pi^{RR1}$  in  $T_1$  for the node  $s_j$  since  $\pi^{RR2}$  schedules the node later than  $\pi^{RR1}$  does. By Remark 1, if a node  $s_j$  cannot send data in  $T_1$  and can send in  $T_2$ , then it would certainly have data to send when it is scheduled in  $T_1 + p$  instead of  $T_2$ . Therefore,  $V_j^{RR1}(T_1) \leq V_j^{RR2}(T_2) \leq V_j^{RR1}(T_1 + p)$  for  $\forall s_j \in S_m$ .

This means that giving each node one more TS, any RR policy can achieve maximum throughput achieved by most efficient RR. In other words, the least efficient RR can achieve the throughput of the most efficient RR by continuing only one period more. Note that since  $U_p$  is the nodes scheduled last under a RR policy, they achieve maximum throughput which can be achieved under RR policy by Remark 1. Therefore, the least efficient RR uses only  $m - k$  TSs more than other RR policies to guarantee same throughput. By using the extra  $m - k$  TSs which the least efficient RR used, the most efficient RR policy can have throughput  $m - k$  more than it has. By giving an example for this situation, Theorem 3 will be proved for this case. Considering the last period  $[T + 1, T + p]$ , the worst performance of RR occurs when the set of nodes  $U_i$  can transmit no data in TS  $T + i$ ; however, they get ready for transmission ( $B_j(t) \geq 1$ ) in TS  $T + i + 1$ . Since there is no next TS for  $U_p$ , nodes of  $U_p$  cannot improve their battery states. Therefore,

the throughput difference is determined by nodes  $s_j \in S_m - U_p$ . Since  $|S_m - U_p| = m - k$ , the difference is  $m - k$ . This concludes the proof.

**Case 3:** Assume that  $N = sp + c$  where  $0 < c < p$ ,  $s \in Z$ . In case 2, it is shown that the maximum difference is  $m - k$  in TS  $sp$ . In the interval  $[sp + 1, sp + c]$ ,  $S_c = \bigcup_{i=1}^c U_i$  is scheduled. For the nodes  $s_j \in U_i \subset S_c$ ,  $B_j(t) \geq 1$  in TS  $sp$ . If  $B_j(sp + i) \geq 2 \forall s_j \in U_i \subset S_c$ , the throughput difference remains as  $m - k$ . Unless  $B_j(sp + i) \geq 2 \forall s_j \in U_i \subset S_c$ , the difference remains same or decreases depending on other nodes  $s_j \in S_m - S_c$ . Hence, it is proved.

## V. OPTIMAL OMNISCIENT POLICIES

For the EH scheduling problem, [18],[20] propose RR-based Myopic Policy (MP) and prove that the MP is optimal for certain specific cases. However, Theorem 2 and Theorem 3 state that RR-based policies with quantum=1 (including the MP in [18],[20]) become suboptimal when  $D_i^{(T)} > 1$  for some sensor  $i$  although an 100% efficient policy exists ( $D^{(T)} \leq 1$ ). The EH scheduling problem resembles a simplified unicast switch scheduling problem. Like unicast switch scheduling problems, this problem has input queues (energy queues) and feasible activation sets are such that at most  $k$  users are scheduled. Different from usual unicast switch scheduling problem setups, buffer (battery) states are not known in this problem; therefore, switch scheduling policies that assume the availability of state information cannot be applied directly. However, these provide intuition for finding an *omniscient scheduling* policy (i.e. one which knows the current battery states) for the EH scheduling problem.

For unicast switch scheduling problems, the following approaches are well-known: Maximum Size Matching (MSM) and Maximum Weight Matching (MWM) [23], [24]. Maximum Size Matching selects in each TS an activation set with the max number of nonempty queues. On the contrary, Maximum Weight Matching also respects queue size not only whether queues are empty or not. Maximum Size Matching may sometimes cause starvation due to head-of-line (HOL) blocking which limits its throughput to below 100% in some cases [23] and [25]. On the other hand, Maximum Weight Matching policies always guarantee 100% throughput for all admissible traffic (with analogy,  $D^{(T)} \leq 1$  in our problem) and two MWM algorithms are offered to achieve 100% throughput in [23]. However, due to lower computational complexity, MSM policies are sometimes preferred [26].

Different from unicast switch scheduling problems, there is no preferred output for packets in

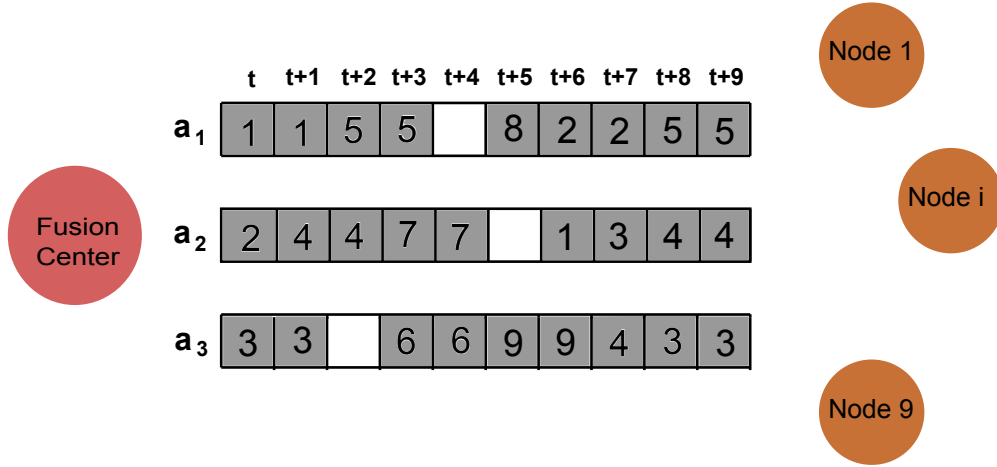


Fig. 2. An example scheduling table kept by the fusion center (FC) for  $m = 9$ ,  $k = 3$  during the interval  $[t, t + 9]$ . Dark colored TSs represent busy slots labeled by node ID using the slot, and the white ones represent idle ones. 3 of 30 slots are idle even under an optimum omniscient policy (UP) ( $D^{(T)} = 0.9$ ). UP allocates the slots in order to leave the least number of slots idle: resembling a Tetris game. Note that each node can use only one channel at a time.

the EH scheduling problem. All  $k$  lines correspond to the same output port to which any packet can be sent. This implies that HOL blocking does not occur in the EH scheduling problem, so both MSM and MWM will provide 100% throughput in our problem. Due to lower computational complexity, MSM is preferable. To find an omniscient policy for the EH scheduling problem, we assume that FC knows whether each node can transmit data or not in any TS. With this knowledge, there is no unique optimal omniscient policy for this problem. We shall concentrate on one optimal policy which provides intuition to find a near optimal, nonomniscient, online policy later.

To find such a policy, we map the problem onto a variation of block-packing game Tetris. A different Tetris model which we are inspired by was previously used in multicast switch scheduling problems [27] and [28]. In this model, packets from same input are sent to different output ports. In our case, different from the Tetris model of [27] and [28], the packets from same input are sent to same output port if the input is scheduled to transmit data to that output port in our model (The model is shown in Figure 2.). That is the critical point which provide us intuition to find a near optimum, nonomniscient, online policy in the next section.

Based on our tetris model, we propose an omniscient, optimum scheduling policy, *Uniformizing Policy (UP)* for all admissible EH process. Considering nonuniform EH processes at all  $m$

nodes, UP uses the empty output ports to schedule the nodes in each TS. If there are some nodes which are scheduled in previous TS but does not have enough energy to transmit data in current TS, UP schedules new nodes. By scheduling new nodes, UP prevents output ports to remain idle and balances the load in each of  $k$  output ports. Hence, UP *uniformizes* the nonuniform EH processes of  $m$  nodes such that all packets are scheduled in each of  $k$  output port almost equally. By this almost equal partition of the packets sent by nodes, UP makes *uniformization* and provides 100% throughput under all admissible uniform and non-uniform EH processes.

The operation of UP is summarized below:

1. Order the nodes arbitrarily and use this order throughout problem horizon.
2. Schedule the first  $k$  nodes in the ordering that have enough energy to transmit a packet.
3. At the beginning of the next TS, check the  $k$  nodes that were just scheduled. Replace those without energy to transmit a packet with new ones, respecting the initial order. If less than  $k$  nodes with enough energy can be found, schedules those nodes only.
4. Continue in a cyclic way.

## VI. A NEAR-OPTIMAL ONLINE POLICY

### A. Uniformizing Random Ordering Policy (UROP)

Assuming that all EH process is known in previous section, an optimal omniscient solution is proposed for the EH scheduling problem. However, the battery states of the nodes are not known in the exact EH scheduling problem. Therefore, we propose a near-optimal online scheduling policy by using Lemma 1 (stated below) for all admissible EH processes  $D^{(T)} < 1$ .  $D^{(T)} < 1$  means that there exists always idle TSs over a problem horizon even if an optimal policy is applied. Lemma 1 states that if a scheduled node cannot transmit data in TS  $t$ , an 100% efficient policy is applied to that node until TS  $t$ . Therefore, we propose UROP which uses the idle TSs to determine battery state of the scheduled nodes (whether a node has enough energy to transmit data or not).

Since EH processes are completely unpredictable for some EH sources [1], [3], UROP orders the nodes randomly before starting to schedule them.

UROP operates as below:



	t	t+1	t+2	t+3	t+4	t+5	t+6	t+7	t+8	t+9	t+10	t+11	t+12	t+13	t+14	t+15	t+16
$a_1$	2	2		5		7	7		1	1		3		6	6		1
$a_2$		3		6	6	6		8	8		2		5	5		7	7
$a_3$	1		4	4	4	4	4	4	4	4	4	4	4	4	4		8

Fig. 3. An example scheduling table kept by the fusion center (FC) for  $m = 8, k = 3$  during the interval  $[t, t + 16]$ . Dark colored TSs represent busy slots labeled by node ID, and the white ones represent idle ones. Node 4 behaves as an elephant node since it does not idle within a round continues transmission from  $t + 2$ , until  $t + 14$ . Note that it has already transmitted data in  $t + 12$  when it is next supposed to be scheduled.

1. Schedule the first  $k$  nodes according to initially determined random order.
2. If a scheduled node transmits data to FC in that TS, then it continues to be scheduled.
3. Otherwise, FC starts to schedule the nodes which have highest priority in the cyclic random order instead of the leaving ones.

To schedule all nodes once, FC uses  $m$  nodes to complete a period (all nodes are scheduled once). As  $D \rightarrow 1$ , the ratio of idle TSs over whole problem horizon decreases. The algorithm, UROP, whose operation is described above is hence an adaptive and near optimal policy. In this section, the efficiency of UROP is investigated by assuming that no node behaves as an *elephant node* (defined below). In section VII, it is shown that UROP is asymptotically optimal over infinite horizon for all admissible EH processes.

*Definition 1 (Elephant node):* If the node who is next in line for selection by the FC happens to be already transmitting continuously since its last selection, the node is said to behave as an *elephant node* between the previous selection (scheduling) time and the current selection time. In this case, FC selects the next node to schedule for one of the empty channels and the elephant node continues to transmit on its assigned channel as before. Figure 3 represents an elephant node.

### B. Efficiency of UROP in Finite Horizon Case

In this part, the efficiency of UROP is investigated in general case of EH process. First, several lemmas are stated and proved. Then, they are used to prove Theorem 4 and Theorem 5.

**Lemma 1 (Partial Optimality):** *If  $B_i(t) < 1$  for a sensor  $s_i$  at the end of TS  $t$ , an optimal policy has been applied for sensor  $s_i$  and efficiency is 100% for sensor  $s_i$  up to  $t$ .*

**Proof of Lemma 1:** The number of data packets which could be sent by sensor  $s_i$  with the remaining energy in TS  $t$  is  $\lfloor B_i(t) \rfloor$ . Since  $B_i(t) < 1$ ,  $\lfloor B_i(t) \rfloor = 0$ . By TS  $t$ ,  $s_i$  has transmit all data which could be sent with  $E_i^{tot}(t)$ , and efficiency is 100% for node  $s_i$  until TS  $t$ .

**Remark 2:** *If  $E_i^{tot}(t)$  is the total amount of harvested energy in sensor  $s_i$  until TS  $t$  and  $V_i^{opt}(t)$  is the number of packets (throughput) which could be sent by sensor  $s_i$  until TS  $t$  under  $\pi^{opt}$ ,  $V_i^{opt}(t) = \lfloor E_i^{tot}(t) \rfloor$ . Recall that  $\pi^{opt} = \pi^{fe}$  for  $0 \leq D \leq 1$ .*

Now, we will define some new parameters which will be used in Lemma 2, Lemma 3, Lemma 4 and Theorem 4. Remember that  $A_k = \{a_1, a_2, \dots, a_k\}$  is the set of mutually orthogonal channels of FC.  $\gamma_l^j$  is the  $l^{th}$  idle TS for  $a_j$ , the  $j^{th}$  channel of the FC. In this TS  $\gamma_l^j$ , FC drops a node using  $j^{th}$  channel and start to schedule another node in same channel.  $\forall \gamma_l^j \in T_I^{(j)}$  Let's denote by  $T_I^{(j)}$  the idle TSs for  $a_j$ , the  $j^{th}$  channel of the FC. In these TSs, FC drops some of the  $k$  nodes and starts to schedule other nodes in their place.  $T_I$  is the set which consists of all pairs  $(a_j, \gamma_l^j)$  where  $a_j \in A_k, \gamma_l^j \in T_I^{(j)}$ . Figure 4 represents the pairs (idle TSs in 2-dimension).

Let's denote by  $\xi_i^{(f)}$  and  $\xi_i^{(f-1)}$  the idle TSs when FC starts to schedule node  $s_i$  for the last time and for the second last time, respectively. If  $a_u \in A_k$  and  $\gamma_v^u \in T_I^{(u)}$ ,  $F_1$  and  $F_2$  are the set of all pairs  $(a_u, \gamma_v^u)$  such that  $\gamma_v^u = \xi_i^{(f)}$  for a  $s_i \in S_m$  and the set of all pairs  $(a_u, \gamma_v^u)$  such that  $\gamma_v^u = \xi_i^{(f-1)}$  for a  $s_i \in S_m$ . There are  $m$  nodes so  $|F_1| = |F_2| = m$ . If  $a_p \in A_k$  and  $\gamma_q^p \in T_I^{(p)}$ ,  $G_1$  is the set of all pairs  $(a_p, \gamma_q^p)$  such that  $\gamma_q^p \neq \xi_i^{(f)}$  for  $s_i \in S_m$ . Moreover,  $G_2$  is the set of all pairs  $(a_p, \gamma_q^p)$  such that  $\gamma_q^p \neq \xi_i^{(f)}$  and  $\gamma_q^p \neq \xi_i^{(f-1)}$  for  $s_i \in S_m$ . In other words,  $G_1 = T_I - F_1$  and  $G_2 = T_I - (F_1 \cup F_2)$ .

**Lemma 2:** *If  $(a_u, \gamma_v^u) \in (F_1 \cup F_2)$ ,*

- i. There exists no  $(a_p, \gamma_q^p) \in G_1$  such that  $\gamma_q^p \neq \gamma_v^u$  for  $\exists (a_u, \gamma_v^u) \in F_1$*
- ii. There exists no  $(a_p, \gamma_q^p) \in G_2$  such that  $\gamma_q^p \neq \gamma_v^u$  for  $\exists (a_u, \gamma_v^u) \in (F_1 \cup F_2)$*

**Proof of Lemma 2:**

**Part i:** Assume that there is a pair  $(a_p, \gamma_q^p) \in G_1$  such that  $\gamma_q^p \neq \gamma_v^u$  for  $\exists (a_u, \gamma_v^u) \in F_1$ . Since  $\gamma_q^p \neq \xi_i^{(f)}$  for  $\forall s_i \in S_m$ , the node  $s_r$  which is selected by the FC in TS  $\xi_q^p$  will be selected by the FC at least once more ( $\gamma_q^p < \xi_r^{(f)}$ ). According to UROP, a node  $s_r$  which is selected in TS  $T_1$  cannot be selected by the FC in TS  $T_2$  unless  $\forall s_i \in S_m - s_r$  are selected in the interval  $[T_1, T_2]$ . Since  $\gamma_q^p > \gamma_v^u = \xi_i^{(f)}$  for  $\exists s_i$ , these nodes cannot be selected by the FC in  $[\gamma_q^p, \xi_j^{(f)}]$ . Therefore,

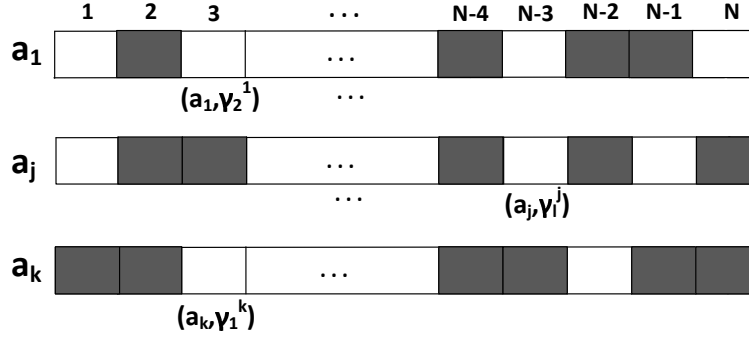


Fig. 4. An example scheduling table kept by the fusion center (FC) for all  $k$  channels over problem horizon  $N$ . Dark colored TSs represent busy slots, and the white ones represent idle ones.

there exists no  $(a_p, \gamma_q^p) \in G_1$  such that  $\gamma_q^p \neq \gamma_v^u$  for  $\exists (a_u, \gamma_v^u) \in F_1$

**Part ii:**  $G_1 = T_I - F_1$  and  $G_2 = T_I - (F_1 \cup F_2) = (T_I - F_1) - F_2$ . Replacing  $T_I - F_1$  and  $F_2$  with  $T_I$  and  $F_1$ , respectively, in case 1, we can said that there exists no  $(a_p, \gamma_q^p) \in G_2$  such that  $\gamma_q^p > \gamma_v^u$  for  $\exists (a_u, \gamma_v^u) \in F_2$ . By case 1 of lemma 2, there exists no  $(a_p, \gamma_q^p) \in F_2$  such that  $\gamma_q^p > \gamma_v^u$  for  $\exists (a_u, \gamma_v^u) \in F_1$ . Therefore, there exists no  $(a_p, \gamma_q^p) \in G_2$  such that  $\gamma_q^p \neq \gamma_v^u$  for  $\exists (a_u, \gamma_v^u) \in (F_1 \cup F_2)$ .

**Lemma 3:** If  $\zeta_j^{(f)}$  is the idle TS when FC stops to schedule node  $s_j$  for the last time and  $L$  is the set of the idle TSs  $\zeta_j^{(f)}$ ,  $L \subset (F_1 \cup F_2)$ .

**Proof of Lemma 3:** Recall that  $(F_1 \cup F_2) \subseteq T_I$ . It can be said that FC starts to schedule a node iff it leaves (stops to schedule) another node.  $(F_1 \cup F_2)$  includes two consecutive time (the last and second last time) when FC starts to schedule a node for all nodes. Assume that FC schedules a node  $s_i$ . Unless FC stops to schedule the node  $s_i$ , it cannot start to schedule the node  $s_i$  again. Therefore,  $(F_1 \cup F_2)$  includes at least one departure time for each node. Since  $(F_1 \cup F_2)$  includes the latest  $2m$  idle TSs and at least one departure time for each node,  $\zeta_j^{(f)} \in (F_1 \cup F_2)$  for  $\forall s_j$ . Hence,  $L \subset (F_1 \cup F_2)$ .

Now, we write Lemma 4 and Lemma 5 which will help us find the lower bound of efficiency.

**Lemma 4:** Assume that  $S_k \subset S_m$  is the set of  $k$  nodes which are scheduled last by the FC in problem horizon  $N$ , each node  $s_j \in S_k$  transmits  $(N - \xi_j^{(f)})$  data in the interval  $[\xi_j^{(f)} + 1, N]$ .

**Proof of lemma 4:** If  $S_k \subset S_m$  is the set of last  $k$  nodes which are scheduled by the FC in problem horizon  $N$ , there will be no other selection so no idle TS until deadline of problem horizon  $N$ . Since each node can transmit at most one packet in each TS, each node  $s_j \in S_k$

transmits  $(N - \xi_j^{(f)}) \times 1 = (N - \xi_j^{(f)})$  data in the interval  $[\xi_j^{(f)} + 1, N]$ .

**Lemma 5:** Assume that  $T_1, T_2 \in (0, N]$  and  $T_1 < T_2$ . If  $s_i$  is not scheduled in interval  $[T_1, T_2]$ ,  $V_i^{opt}(T_1) \leq V_i^{opt}(T_2)$  where  $V_i^{opt}(t)$  is the number of packets (throughput) which could be sent by sensor  $s_i$  until TS  $t$  under optimal policy  $\pi^{opt}$ .

**Proof of lemma 5:** By Remark 2,  $V_i^{opt}(T_1)$  and  $V_i^{opt}(T_2)$  can be written as below:

$$V_i^{opt}(T_1) = \lfloor E_i^{tot}(T_1) \rfloor \text{ and } V_i^{opt}(T_2) = \lfloor E_i^{tot}(T_2) \rfloor \quad (24)$$

From Remark 1 (No battery leakage), the inequality below is written for  $\forall s_i \in S_m$

$$E_i^{tot}(T_1) \leq E_i^{tot}(T_2) \quad (25)$$

$$\lfloor E_i^{tot}(T_1) \rfloor + (E_i^{tot}(T_1))^+ \leq \lfloor E_i^{tot}(T_2) \rfloor + (E_i^{tot}(T_2))^+ \quad (26)$$

By putting (24) into (26),

$$V_i^{opt}(T_1) + (E_i^{tot}(T_1))^+ \leq V_i^{opt}(T_2) + (E_i^{tot}(T_2))^+ \quad (27)$$

By using (27), it is shown that  $V_i^{opt}(T_1) \leq V_i^{opt}(T_2)$  is possible; however,  $V_i^{opt}(T_1) > V_i^{opt}(T_2)$  is not possible. There are three cases as below:

- i)  $V_i^{opt}(T_1) = V_i^{opt}(T_2) \Rightarrow (E_i^{tot}(T_1))^+ \leq (E_i^{tot}(T_2))^+$  since  $E_i^{tot}(T_1) \leq E_i^{tot}(T_2)$
- ii)  $V_i^{opt}(T_1) < V_i^{opt}(T_2) \Rightarrow E_i^{tot}(T_1) < E_i^{tot}(T_2)$  since  $0 \leq (E_i^{tot}(T_1))^+, (E_i^{tot}(T_2))^+ < 1$  and  $V_i^{opt}(T_1), V_i^{opt}(T_2) \in \mathbb{Z}$
- iii)  $V_i^{opt}(T_1) > V_i^{opt}(T_2) \Rightarrow E_i^{tot}(T_1) > E_i^{tot}(T_2)$  since  $0 \leq (E_i^{tot}(T_1))^+, (E_i^{tot}(T_2))^+ < 1$  and  $V_i^{opt}(T_1), V_i^{opt}(T_2) \in \mathbb{Z}$  This situation contradicts with (26).

Hence,  $E_i^{tot}(T_1) \leq E_i^{tot}(T_2)$  for  $\forall s_i \in S_m$  and  $V_i^{opt}(T_1) \leq V_i^{opt}(T_2)$  for  $\forall s_i \in S_m$ .

**Theorem 4 (Efficiency Bounds of UROP):** Last departure time of the node  $s_j = s_0$  which satisfies  $\zeta_j^{(f)} \leq \zeta_i^{(f)}$  for  $\forall s_i \in S_m - \{s_j\}$  is denoted by  $\zeta_j^{(f)} = T_0$ . In problem horizon  $N$ , the efficiency of UROP is bounded as below:

$$1 - \frac{k(N - T_0)}{\sum_{i=1}^m V_i^{opt}(N)} \leq \eta_{UROP} \leq 1 \quad (28)$$

where  $V_i^{opt}(N)$  is the number of packets (throughput) which could be transmitted by sensor  $s_i$  until TS  $t$  (included) under optimal policy  $\pi^{opt}$ .

**Proof of Theorem 4:**  $V_i(t)$  is the number of packets (throughput) which have been sent by sensor  $s_i$  until TS  $t$ .  $V_i^{(f)}(\zeta_i^{(f)})$  is the number of packets (throughput) which could be transmitted by sensor  $s_i$  in the interval  $[\zeta_i^{(f)}, N]$ .

$V_i^{opt} = V_i(\zeta_i^{(f)}) + V_i^{(f)}(\zeta_i^{(f)})$  can be written for  $\forall s_i \in S_m$ . By Lemma 1,  $V_i(\zeta_i^{(f)})$  is the throughput in TS  $\zeta_i^{(f)}$  until when an optimum policy  $\pi^{opt}$  is applied to node  $s_i$ . Therefore,  $V_i^{(f)}(\zeta_i^{(f)})$  is the only factor for throughput loss of node  $s_i \in S_m - S_k$ . For  $s_i \in S_k$ , the throughput loss by  $V_i^{(f)}(\zeta_i^{(f)})$  decreases by  $(N - \xi_i^{(f)})$  by Lemma 4.

Hence, the efficiency of UROP in problem horizon  $N$  can be written as below:

$$\begin{aligned} \eta_{UROP} &= \frac{\sum_{i=1}^m V_i(\zeta_i^{(f)}) + \sum_{s_i \in S_k} (N - \xi_i^{(f)})}{\sum_{i=1}^m V_i^{opt}(N)} \\ &= \frac{\sum_{i=1}^m V_i(\zeta_i^{(f)}) + \sum_{s_i \in S_k} (N - \xi_i^{(f)})}{\sum_{i=1}^m V_i(\zeta_i^{(f)}) + \sum_{i=1}^m V_i^{(f)}(\zeta_i^{(f)})} \end{aligned} \quad (29)$$

By using Lemma 4, the term  $\sum_{s_i \in S_k} (N - \xi_i^{(f)})$  is added to numerator in (29) since  $s_i \in S_k$  are not considered to be left by the FC in TS  $N$ . It is assumed that  $\zeta_i^{(f)} < N$  for  $\forall s_i \in S_k$ .

By (29), we can upper bound  $\eta_{UROP}$ .

#### ***i. Upper bound for efficiency of UROP***

Efficiency of a policy cannot be more than 100% ( $\eta \leq 1$ ). From (29),  $\eta_{UROP} = 1$  only if the equality (30) is satisfied:

$$\sum_{s_i \in S_k} (N - \xi_i^{(f)}) = \sum_{i=1}^m V_i^{(f)}(\zeta_i^{(f)}) \quad (30)$$

(30) comes true only if (31) is satisfied:

$$V_i^{(f)}(\zeta_i^{(f)}) = \begin{cases} 0 & \text{if } s_i \in S_m - S_k \\ (N - \xi_i^{(f)}) & \text{if } s_i \in S_k \end{cases} \quad (31)$$

If sensors harvest energy such that (31) is satisfied,  $\eta_{UROP} = 1$ . Therefore, upper bound of  $\eta_{UROP}$  is 100%, namely,  $\eta_{UROP} \leq 1$ .

By (29), let's find the lower bound of  $\eta_{UROP}$ .

#### ***ii. Lower bound for efficiency of UROP***

The inequalities below can be written for a long problem horizon  $N$ .

$$\sum_{s_i \in S_k} (N - \xi_j^{(f)}) \leq \sum_{i=1}^m V_i^{(f)}(\zeta_i^{(f)}) < \sum_{i=1}^m V_i(\zeta_j^{(f)}) < \sum_{i=1}^m V_i^{opt}(N) \quad (32)$$

To find the lower bound of  $\eta_{UROP}$ , we will define a loss function  $V_{loss}$  in (33) according to (29) and maximize  $V_{loss}$  by considering the worst case.

$$V_{loss} = \sum_{i=1}^m V_i^{(f)}(\zeta_i^{(f)}) - \sum_{s_i \in S_k} (N - \xi_i^{(f)}) \quad (33)$$

In (33),  $V_{loss}$  can be maximized by minimizing  $\sum_{s_i \in S_k} (N - \xi_i^{(f)})$ . Since  $\xi_i^{(f)} \leq N$  for  $\forall s_i \in S_k$ ,  $\sum_{s_i \in S_k} (N - \xi_i^{(f)}) \geq 0$ . This occurs only if  $\xi_i^{(f)} = N$  for  $\forall s_i \in S_k$ .

By Equation (5),  $V_i^{(f)}(\xi_i^{(f)}) = \left\lfloor B_i(\xi_i^{(f)}) + \sum_{t=\xi_i^{(f)}+1}^N E_i^h(t) \right\rfloor$ .

Since  $\sum_{s_i \in S_k} (N - \xi_i^{(f)}) \geq 0$ ,  $V_{loss}$  is maximized if  $\sum_{s_i \in S_k} (N - \xi_i^{(f)}) = 0$ . Then (33) converts into (34).

$$V_{loss} = \sum_{s_i \in S_m} V_i^{(f)}(\zeta_i^{(f)}) \quad (34)$$

We denote by  $S_k^{(lf)}$  the set of  $k$  nodes which satisfy  $\zeta_i^{(f)} \leq \zeta_j^{(f)}$  for  $\forall s_i \in S_k^{(lf)}$  and  $s_i \in S_m - S_k^{(lf)}$ .  $V_{loss}$  can be written as follows:

$$V_{loss} = \sum_{s_i \in S_m} V_i^{(f)}(\zeta_i^{(f)}) = \sum_{s_i \in S_k^{(lf)}} V_i^{(f)}(\zeta_i^{(f)}) + \sum_{s_i \in S_m - S_k^{(lf)}} V_i^{(f)}(\zeta_i^{(f)}) \quad (35)$$

While maximizing  $V_{loss}$ , (36) must be considered for  $\forall s_i \in S_m$  from Theorem 1.

$$V_{av}^{(\zeta_i^{(f)})}(t) \leq \frac{k(N - T)}{m}, \forall s_i \in S_m \quad (36)$$

From (36), FC can accumulate maximum  $k$  data. This scheduling capacity can be achieved if there is an energy harvesting process such that  $k$  of  $m$  nodes can transmit 1 data in each TS and the remaining nodes can transmit no data.

In this case,  $V_{loss}$  becomes maximum when each sensor  $s_i \in S_k^{(lf)}$  harvests 1 unit energy and the other sensors  $s_i \in S_m - S_k^{(lf)}$  harvest almost no energy in each TS. It can be shown as below in (37). By putting (34) in (35), (37) can be written as below:

$$V_{loss} = \sum_{s_i \in S_k^{(lf)}} \left\lfloor B_i(\zeta_i^{(f)}) + \sum_{t=\zeta_i^{(f)}+1}^N E_i^h(t) \right\rfloor + \sum_{s_i \in S_m - S_k^{(lf)}} \left\lfloor B_i(\zeta_i^{(f)}) + \sum_{t=\zeta_i^{(f)}+1}^N E_i^h(t) \right\rfloor \quad (37)$$

Last departure time of the node  $s_j = s_0$  which satisfies  $\zeta_i^{(f)} \leq \zeta_j^{(f)}$  for  $\forall s_i \in S_m - \{s_j\}$  is denoted by  $\zeta_j^{(f)} = T_0$ . By using Lemma 5, we write an upper bound for  $V_{loss}$  as below in (38):

$$V'_{loss} = \sum_{s_i \in S_k^{(lf)}} \left\lfloor B_i(T_0) + \sum_{t=T_0+1}^N E_i^h(t) \right\rfloor + \sum_{s_i \in S_m - S_k^{(lf)}} \left\lfloor B_i(T_0) + \sum_{t=T_0+1}^N E_i^h(t) \right\rfloor \geq V_{loss} \quad (38)$$

To maximize  $V_{loss}$ , maximizing  $V'_{loss}$  will be enough so take  $V_{loss} = V'_{loss}$ . To satisfy this equality, we assume that  $T_i = T_0$  for  $s_i \in S_k^{(lf)}$ . By using (36), the inequality (39) can be written

$$V_{av}^{(T_0)}(t) = \frac{1}{m} \sum_{s_i \in S_m} V_i^{(T_0)}(t) \leq \frac{k(N - T_0)}{m} \quad (39)$$

$$V_{tot}^{(T_0)}(t) = \sum_{s_i \in S_m} V_i^{(T_0)}(t) \leq k(N - T_0) \quad (40)$$

$$V_{tot}^{(T_0)}(t) = \sum_{s_i \in S_k^{(lf)}} V_i^{(T_0)}(t) + \sum_{s_i \in S_m - S_k^{(lf)}} V_i^{(T_0)}(t) \leq k(N - T_0) \quad (41)$$

$$\begin{aligned} V_{tot}^{(T_0)}(t) &= \sum_{s_i \in S_k^{(lf)}} \left[ V_i^{(T_0)}(t) - V_i^{(f)}(\zeta_i^{(f)}) + V_i^{(f)}(\zeta_i^{(f)}) \right] \\ &+ \sum_{s_i \in S_m - S_k^{(lf)}} \left[ V_i^{(T_0)}(t) - V_i^{(f)}(\zeta_i^{(f)}) + V_i^{(f)}(\zeta_i^{(f)}) \right] \leq k(N - T_0) \end{aligned} \quad (42)$$

Since  $T_0 = \zeta_i^{(f)}$  for all  $s_i \in S_k^{(lf)}$ ,  $V_i^{(T_0)}(t) - V_i^{(f)}(\zeta_i^{(f)}) = 0$  for all  $s_i \in S_k^{(lf)}$ . Hence, the inequality converts into (43),

$$\begin{aligned} V_{tot}^{(T_0)}(t) &= \sum_{s_i \in S_k^{(lf)}} V_i^{(f)}(\zeta_i^{(f)}) + \sum_{s_i \in S_m - S_k^{(lf)}} \left[ V_i^{(T_0)}(t) - V_i^{(f)}(\zeta_i^{(f)}) \right] \\ &+ \sum_{s_i \in S_m - S_k^{(lf)}} \left[ V_i^{(f)}(\zeta_i^{(f)}) \right] \leq k(N - T_0) \end{aligned} \quad (43)$$

By using (35) and (38) for  $V'_{loss}$  in (43), (44) can be written as below:

$$V_{tot}^{(T_0)}(t) = V'_{loss} + \sum_{s_i \in S_m - S_k^{(lf)}} \left[ V_i^{(T_0)}(t) - V_i^{(f)}(\zeta_i^{(f)}) \right] \leq k(N - T_0) \quad (44)$$

Since  $T_0 \leq \zeta_i^{(f)}$  for all  $s_i \in S_m - S_k^{(lf)}$ , by Lemma 5, (45) can be written

$$V_i^{opt}(T_0) \leq V_i^{opt}(\zeta_i^{(f)}), \forall s_i \in S_m - S_k^{(lf)} \quad (45)$$

$$V_i^{tot}(N) - V_i^{opt}(T_0) \geq V_i^{opt}(N) - V_i^{opt}(\zeta_i^{(f)}), \forall s_i \in S_m - S_k^{(lf)} \quad (46)$$

$$V_i^{(T_0)}(T_0) \geq V_i^{(f)}(\zeta_i^{(f)}), \forall s_i \in S_m - S_k^{(lf)} \quad (47)$$

By (44), to maximize  $V'_{loss}$ ,  $\sum_{s_i \in S_m - S_k^{(lf)}} \left[ V_i^{(T_0)}(t) - V_i^{(f)}(\zeta_i^{(f)}) \right]$  should be minimized. By (47),  $V_i^{(T_0)}(t) - V_i^{(f)}(\zeta_i^{(f)}) \geq 0$ . If  $V_i^{(T_0)}(t) - V_i^{(f)}(\zeta_i^{(f)}) = 0$ , (44) converts into (48),

$$V_{tot}^{(T_0)}(t) = V'_{loss} \leq k(N - T_0) \quad (48)$$

By using (38) and (48),

$$V_{loss} \leq k(N - T_0) \quad (49)$$



By (29), efficiency of UROP can be written as below:

$$\eta_{UROP} = 1 - \frac{\sum_{i=1}^m V_i^{(f)}(\zeta_i^{(f)}) - \sum_{s_i \in S_k} (N - \xi_i^{(f)})}{\sum_{i=1}^m V_i(\zeta_i^{(f)}) + \sum_{i=1}^m V_i^{(f)}(\zeta_i^{(f)})} \quad (50)$$

Recall that  $\sum_{s_i \in S_k} (N - \xi_i^{(f)}) \geq 0$ . From (32),  $\sum_{i=1}^m V_i^{(f)}(\zeta_i^{(f)}) \ll \sum_{i=1}^m V_i(\zeta_i^{(f)})$ ,

$$\eta_{UROP} \geq 1 - \frac{V_{loss}}{\sum_{i=1}^m V_i^{opt}(N)} \quad (51)$$

$$\geq 1 - \frac{\sum_{i=1}^m V_i^{(f)}(\zeta_i^{(f)})}{\sum_{i=1}^m V_i^{opt}(N)} \quad (52)$$

$$\geq 1 - \frac{k(N - T_0)}{\sum_{i=1}^m V_i^{opt}(N)} \quad (53)$$

Hence, Theorem 4 is proven and the efficiency of UROP is bounded as below:

$$1 - \frac{k(N - T_0)}{\sum_{i=1}^m V_i^{opt}(N)} \leq \eta_{UROP} \leq 1 \quad (54)$$

*When elephant nodes are present:* Regular nodes scheduled by UROP, give rise to at least one idle TS in a period (frame). However, this does not hold for elephant nodes. If there are nodes that behave as elephant nodes in a period, these do leave any TS empty in that period. Consequently, for these nodes UROP behaves as UP, which does not give up TS to determine the battery states of nodes. Hence, efficiency bounds in Theorem 4 are also valid in case of elephant nodes.

Considering *the worst case* in Theorem 4, we found lower and upper bounds for the efficiency of UROP in terms of parameters.  $k$  is known and  $V_i^{opt}(N)$  can be found for each node  $s_i$  by Remark 2. However, the parameter  $T_0$  cannot be determined unless all details of scheduling in problem horizon is known. Due to the uncertainty of  $T_0$ , Theorem 4 does not give sufficient information about efficiency of UROP. As we mentioned in Section II, expected average reward is a suitable performance measure for the EH scheduling policy over finite or infinite horizon [22]. Considering  $T_0$  (and the other departure times of nodes) as ergodic processes depending on EH processes, we take expectation of the bounds of UROP in Theorem 5. Hence, the bounds of UROP can be determined in expected manner.

**Theorem 5:** For  $0 < D < 1$ , expected efficiency of UROP is bounded as below:

$$1 - \frac{2m}{(1 - D)DNk} \leq E\{\eta_{UROP}\} \leq 1 \quad (55)$$

where  $D, N, m$ , and  $k$  are density, problem horizon length, number of the sensors, number of the orthogonal channels of the FC, respectively.

**Proof of Theorem 5:** By Theorem 4, the efficiency of UROP can be written as below:

$$1 - \frac{k(N - T_0)}{\sum_{i=1}^m V_i^{opt}(N)} \leq \eta_{UROP} \leq 1 \quad (56)$$

$$1 - \frac{k(N - T_0)}{V_{tot}(N)} \leq \eta_{UROP} \leq 1 \quad (57)$$

$$1 - E \left\{ \frac{k(N - T_0)}{V_{tot}(N)} \right\} \leq E \{ \eta_{UROP} \} \leq 1 \quad (58)$$

$$1 - \frac{kE \{ N - T_0 \}}{V_{tot}(N)} \leq E \{ \eta_{UROP} \} \leq 1 \quad (59)$$

We denote by  $\tau_{ar,i}$  and  $\tau_{dep,i}$ , elapsed time between two consecutive selection of same sensor  $s_i$  and elapsed time between two consecutive departure of same sensor  $s_i$ . For long problem horizons,  $E \{ \tau_{ar,i} \} = E \{ \tau_{dep,i} \}, \forall i$ . By Lemma 3,  $L \subset (F_1 \cup F_2)$ . By Lemma 2, none of nodes  $s_i \in S_m - S_k$  can be selected (started to schedule) more than twice by the FC in the interval  $[T_0, N]$ ; therefore,  $E \{ N - T_0 \} < 2E \{ \tau_{ar} \}$ . None of the nodes  $s_i \in S_k$  can be left (stopped to schedule) more than once by the FC in interval  $[\zeta_i^{(f)}, N]$ ; therefore,  $E \{ N - \zeta_i^{(f)} \} < 2E \{ \tau_{dep} \}$ . Hence, (59) is converted into (60):

$$1 - \frac{2kE \{ \tau_{ar} \}}{V_{tot}(N)} \leq E \{ \eta_{UROP} \} \leq 1 \quad (60)$$

Let denote by  $D$  and  $K$ , density during problem horizon  $N$  and the number of orthogonal channels of the FC. By definition of  $D$ ,  $V_{tot}(N) = DNk$ .

$$D = \frac{kE \{ \tau_{ar} \} - m}{kE \{ \tau_{ar} \}} \quad (61)$$

$$E \{ \tau_{ar} \} = \frac{m}{k(1 - D)} \quad (62)$$

$$1 - \frac{2k \frac{m}{k(1-D)}}{DNk} < E \{ \eta_{UROP} \} \leq 1 \quad (63)$$

$$1 - \frac{2m}{(1 - D) DNk} < E \{ \eta_{UROP} \} \leq 1 \quad (64)$$

**Note:** Since  $D = 0$  means no harvested energy in the whole network, it is trivial case and not considered in our calculations.  $D = 1$  means that there is no idle TS if FC apply the 100% efficient policy ( $\pi^{fe}$ ). However, UROP benefits from idle TSs to schedule the sensors. From Theorem 1, no  $\pi^{fe}$  exists for  $D > 1$ . Therefore, we investigate  $0 < D < 1$  in this paper.

## VII. EXTENSION TO THE INFINITE-HORIZON CASE

As in (16) and (22), efficiency of RR-based policies (also MP in [18] and [20]) depend on not only sensor densities  $D$  and  $D^{(T)}$  but also partial sensor densities  $D_i$  and  $D_i^{(T)}$  and cannot improve as problem horizon goes to infinity. Also, it is proved that batteries of nodes for which  $D_i > 1$  and  $D_i^{(T)} > 1$  will overflow over infinite horizon. However, efficiency of UROP in finite horizon case improves as the problem horizon increase and goes to infinity. By Theorem 5 and the relation  $V_{tot}^{opt}(N) = DNk$ , efficiency of UROP is, for  $0 < D < 1$ ,

$$\lim_{N \rightarrow \infty} \left( 1 - \frac{2m}{(1-D)DNk} \right) < \lim_{N \rightarrow \infty} E \{ \eta_{UROP} \} \leq 1 \quad (65)$$

Hence,  $\lim_{N \rightarrow \infty} E \{ \eta_{UROP} \} = 1$ , which shows that UROP is *asymptotically optimal* in the infinite horizon for general EH processes.

## VIII. NUMERICAL RESULTS

In this section, efficiency achieved by RR and UROP policies are compared for independent (Poisson) and correlated (Markovian) EH processes under high density ( $D=0.975$ ) and low density ( $D=0.2$ ) EH processes first. RR and UROP are then compared under a fairness criterion for independent (Poisson) and correlated (Markovian) EH processes under high density ( $D=0.975$ ). Finally, computational complexities of RR, UROP and UP (the omniscient policy proposed in section V) are compared. We focus on the region  $D^{(T)} \leq 1$  so  $\eta_{opt} = \eta_{fe} = 1$ .

To begin with, we compare efficiencies of these policies under both infinite and finite battery assumption for four cases. To create a realistic scenario, we take  $m = 100$ ,  $k = 10$ ,  $N = 2000$  for both policies. We also investigate the efficiency of UROP by taking  $m = 103$  and  $k = 10$ . Note that we compare efficiency of UROP with an arbitrary RR since  $\eta_{RR} \cong \eta_{MP}$  for long problem horizons (Theorem 3). We investigate the efficiencies of both policies under a nonuniform EH process (Both achieve nearly 100% efficiency for uniform EH processes). Nonuniform, high density traffic is formed by taking  $D_i = 3$  for 25 of the nodes and  $D_i = 0.3$  for the remaining ones. Moreover, low density, nonuniform traffic is formed by taking  $D_i = 2.1$  for 5 nodes and  $D_i = 0.1$  for the remaining nodes. Independent EH processes are modelled as Poisson. Markov EH process are modelled by a state space  $M_i = \{0, 1, 2\}$ ,  $\forall s_i$  and a  $3 \times 3$  transition matrix  $P$  such that  $P_{ii} = 0.9 \forall i$  and  $P_{ij} = 0.05$  for  $i \neq j$ . The harvested energy for node  $s_i$  in TS  $t$ ,

$E_i^h(t)$ , is determined by  $M_i$  such that  $E_i^h(t) = D_i \times M_i(t)$  (Note that each transmission requires unit energy.).

In Figure 5 (Low density, independent EH process), UROP has nearly 100% efficiency whereas RR has approximately 80% efficiency. In Figure 6 (High density, independent EH process), UROP continues to attain nearly 100% efficiency whereas the efficiency of RR has dropped below 50%. This is an expected result since Theorem 2 states that as the number of nodes s.t.  $D_i > 1$  increases, efficiency of RR decreases. By (13), efficiency of RR is expected to be  $\eta_{RR} = 48.7\%$  and  $\eta_{RR} = 72.5\%$  for the low and high density EH process, respectively.

In Figure 7 (Low density, Markov EH process), UROP has nearly 100% efficiency whereas RR has nearly 70% efficiency. In Figure 8 (High density, Markov EH process), UROP has nearly 100% efficiency whereas RR has nearly 50% efficiency. When the EH process has memory, we observe similar results, except that the performance of RR drops further. The efficiency of UROP is more robust to memory in harvest process, as compared to RR (Note that  $P_{ii} = 0.9, \forall i$ ).

Considering all four figures, we wish to make three additional remarks. First, the efficiency of UROP converges to 100%  $N \rightarrow \infty$ , as shown in Section VII (UROP is asymptotically optimal). Secondly, efficiency of UROP with a reasonable-sized finite battery  $B_i=50$  is almost same as that with infinite battery. Finally, UROP can achieve nearly 100% throughput both for  $m/k \in \mathbb{Z}$  and  $m/k \notin \mathbb{Z}$  cases, while RR needs  $m/k \in \mathbb{Z}$  assumption for optimality. We conclude that UROP is more adaptive and efficient than RR (and MP proposed in [18], [20] by Theorem 3).

In addition to throughput, the performances of RR and UROP are also compared in terms of fairness, which is often an important issue for scheduling policies. We apply Jain's Fairness index [29],  $f(x) = \frac{[\sum_{i=1}^m x_i(t)]^2}{m \sum_{i=1}^m x_i^2(t)}$  where  $x_i(t)$  is the  $i^{th}$  user allocation up to TS  $t$ . Adopting the *proportionate progress* (P-fairness) criterion in [30], we scale the allocation  $x_i(t) = \frac{V_i(t)}{V_i^{opt}(t)}$  over users.

RR is usually known as a fair policy since it schedules users periodically. RR is 100% fair for uniform EH processes. However, RR may not be very fair for nonuniform EH processes. In fact, from 16 and 22 the efficiency of RR is expected to be  $FI_{RR} = 89.3\%$  for high density  $D = 0.975$ , nonuniform arrivals. On the other hand, UROP schedules the users proportionally to their loads as well as respecting same or periodically. Consequently, UROP can achieve 100% fairness for general case of EH process. This is evident on Figure 9 and Figure 10. It is also observed that UROP is nearly 100% fair also for  $m/k$  noninteger case.

In addition to throughput and fairness, the policies are compared in terms of computational complexity. RR has complexity  $O(1)$ . Besides achieving almost 100% throughput and 100% fairness for various EH processes, UROP has low-complexity as well. In each TS, UROP checks the  $k$  nodes which are scheduled in previous TS thus it makes only  $k$  computations in each TS. Therefore, computational complexity of UROP is  $O(kN)$ . It is also interesting to compare UROP with UP in terms of computational complexity. UP is an optimal omniscient policy. In each TS, UP checks the  $k$  nodes scheduled in previous TS and looks for replacement nodes if some of the  $k$  nodes cannot transmit data in that TS. Number of computation which UP makes in each TS is between  $k$  and  $m$ . Hence, UP has a computational complexity between  $O(kN)$  and  $O(mN)$ . The results show that to achieve 100% throughput, UP may have complexity  $O(mN)$  whereas UROP may have complexity  $O(kN)$ . This implies that UP may have  $\frac{m}{k}$  times more computation than UROP to achieve the same throughput performance. In other words, UROP achieves the same performance as UP with up to  $\frac{m}{k}$  times lower complexity.

## IX. CONCLUSION

This paper investigated a scheduling problem for a single-hop WSN where a fusion center(FC) schedules a set of EH nodes to receive data from them. FC does not know the instantaneous battery states of nodes. Batteries get recharged according to random Energy Harvesting processes, whose statistics are not available to the FC, and there is no leakage from the batteries. Under an infinite battery capacity assumption, we exhibit a near-optimal online scheduling policy for a broad set of EH processes (Markovian, independent, uniform, nonuniform, etc.)

The scheduling problem is set up as an expected undiscounted reward maximization problem. It is shown that Round Robin (RR) based policies are generally suboptimal(do not guarantee 100% throughput) for nonuniform EH processes. It is also shown that policies proposed in previous literature (namely, myopic policies in [18] and [20] have almost equal efficiency as any other RR policy with quantum=1.

Next, a low-complexity scheduling policy, UROP, is proposed. It is shown that UROP is asymptotically optimal regardless of traffic, in the infinite horizon. Even in the finite horizon, UROP achieves nearly 100% throughput without requiring feedback about battery states of nodes. As this is a type of restless multi-armed bandit problem, the simple self-adapting scheduling technique of UROP could find potential applications in problems other than communication

networks, whenever the performance measure is average reward and the queues store a flexible (time insensitive) resource such as energy.

## REFERENCES

- [1] J. A. Paradiso and T. Starner, "Energy scavenging for mobile and wireless electronics", *IEEE Pervasive Computing*, pp. 18-27, Jan. 2005
- [2] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, "Power management in energy harvesting sensor networks", in *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 6, no. 4, p.32-es, September 2007.
- [3] C. F. Garca-Hernandez, P. H. Ibargengoytia-Gonzalez, J. Garca-Hernandez, J. A. Prez-Daz, "Wireless Sensor Networks and Applications: a Survey", *IJCSNS International Journal of Computer Science and Network Security*, Vol. 7, No. 3, March 2007, pp. 264-273.
- [4] S. Sudevalayam, P. Kulkarni, "Energy Harvesting Sensor Nodes: Survey and Implications," *IEEE Communications Surveys & Tutorials*, vol.13, no.3, pp.443-461, Third Quarter 2011.
- [5] R. E. Bellman, *Dynamic Programming*. Princeton, N.J.: Princeton University Press, 1957
- [6] C. J. Watkins, "Learning from delayed rewards", *Ph.D. dissertation*, University of Cambridge, Psychology Dep., 1989
- [7] L. P. Kaelbling, Michael L. Littman, Andrew W. Moore, "Reinforcement learning: a survey", *Journal of Artificial Intelligence Research*, v.4 n.1, January 1996, pp.237-285
- [8] C. J. Watkins, P. Dayan (1992). "Q-learning". *Machine Learning*, 8 (3), pp. 279-292
- [9] S. Mahadevan, "Average reward reinforcement learning: Foundations, algorithms, and empirical results," *Machine Learning, Special Issue on Reinforcement Learning*, vol. 22, pp. 159-196, 1996
- [10] R.S. Sutton, A.G. Barto, *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 1998
- [11] J. C. Gittins, "Bandit processes and dynamic allocation indices (with discussion)", *J. Roy. Statist. Soc. Series B*, Vol. 41, No.2 (1979) pp. 148-177.
- [12] Whittle, P. (1988). "Restless bandits: Activity allocation in a changing world. In *A Celebration of Applied Probability* J. Gani (Ed.), J. Appl. Prob. 25A 287-298
- [13] C. H. Papadimitriou and J. N. Tsitsiklis, "The complexity of optimal queueing network control", *Math. Oper. Res.*, vol 24, May 1999, pp. 293-305
- [14] S. H. A. Ahmad, M. Liu, T. Javidi, Q. Zhao and B. Krishnamachari, "Optimality of myopic sensing in multi-channel opportunistic Access" *IEEE Trans. Inf. Theory*, vol 55, No. 9, pp. 4040-4050, Sept. 2009.
- [15] S. H. A. Ahmad, L. Mingyan, "Multi-channel opportunistic access: A case of restless bandits with multiple plays," in *Proc. 47th Ann. Allerton Conf. Commun., Contr., Comput.*, Monticello, IL, pp. 1361-1368, Sept. 2009.
- [16] K. Liu and Q. Zhao, "Indexability of restless bandit problems and optimality of Whittle index for dynamic multichannel access," *IEEE Trans. Inf. Theory*, vol 56, no. 11, pp. 5547-5567, Nov. 2010
- [17] A. Hero, D. Castanon, D. Cochran, K. Kastella, *Foundations and Applications of Sensor Management*, Chapter 6, Springer, US, 2007
- [18] P. Blasco, D. Gunduz, and M. Dohler, "Low-Complexity Scheduling Policies for Energy Harvesting Communication Networks", *IEEE International Symposium on Information Theory (ISIT)*, Istanbul, Turkey, July 2013, pp. 1-5
- [19] M. Johnston, E. Modiano, I. Keslassy "Channel Probing in Communication Systems: Myopic Policies Are not Always Optimal", *IEEE International Symposium on Information Theory (ISIT)*, Istanbul, Turkey, July 2013, pp. 1-5

- [20] F. Iannello, O. Simeone, and U. Spagnolini, "Optimality of myopic scheduling and whittle indexability for energy harvesting sensors," in *46th Annual Conference on Information Sciences and Systems(CISS)*, Princeton, NJ, USA, March 2012, pp. 16
- [21] J. Gittins, K. Glazerbrook, R. Weber, *Multi-armed bandit allocation indices*. West Sussex, UK, Wiley, 2011.
- [22] A. Arapostathis, V. S. Borkar, E. Fernandez-gaucherand, M. K. Ghosh, and S. I. Marcus, "Discrete-time controlled Markov processes with average cost criterion: A survey," *SIAM J. Control Optim.*, vol. 31, no. 2, 1993, pp. 2823-44.
- [23] N. McKeown, A. Mekkittikul, V. Anantharam, J. Walrand, "Achieving 100% throughput in an input-queued switch", *IEEE Transactions on Communications*, Vol. 47, No. 8, pp. 1260-1267, August 1999.
- [24] Yanming Shen, S. Panwar, and H. J. Chao, "Design and performance analysis of a practical load-balanced switch," *IEEE Transactions on Communications*, vol. 57, pp. 2420-2429, 2009
- [25] I. Keslassy, R. Z. Shen, and N. McKeown, "Maximum size matching is unstable for any packet switch," *IEEE Communications Letters*, vol. 7, pp. 496-498, Oct. 2003
- [26] A. Mekkittikul and N. McKeown, "A practical scheduling algorithm to achieve 100% throughput in input-queued switches," in *Proceedings of IEEE Infocom*, vol. 2, San Francisco, CA, Apr. 1998, pp. 792-799.
- [27] B. Prabhakar, N. McKeown, and J. Mairesse: "Tetris Models for Multicast Switches", *Proc. of the 30th Annual Conference on Information Sciences and Systems*, Princeton. 1996
- [28] B. Prabhakar, N. McKeown and R. Ahuja, "Multicast scheduling for input-queued switches," *IEEE Journal of Selected Areas Communication*, vol. 15, pp. 855-866, May 1997.
- [29] R. Jain, D.-M. Chiu and W. Hawe, "A Quantitative Measure of Fairness and Discrimination For Resource Allocation in Shared Computer Systems," *Technical Report TR-301, DEC Research Report*, September, 1984
- [30] S. K. Baruah, N. K. Cohen, C. G. Plaxton, and D. A. Varel, "Proportionate progress: A notion of fairness in resource allocation," *Algorithmica*, 15(6):600-625, 1996.



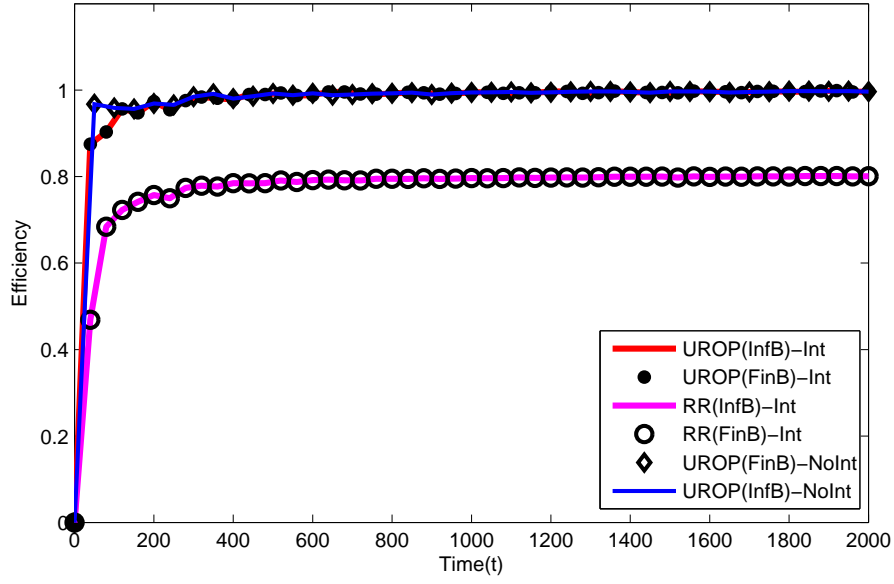


Fig. 5. Efficiencies (ratio of total throughput by a policy to total throughput by optimal policy) of UROP, RR under infinite and finite battery  $B_i=50$  assumptions for independent low density energy arrivals ( $D = 0.2$ ) such that  $m/k \in \mathbb{Z}$ . Efficiency of UROP is also shown for  $m/k$  taking a noninteger value.

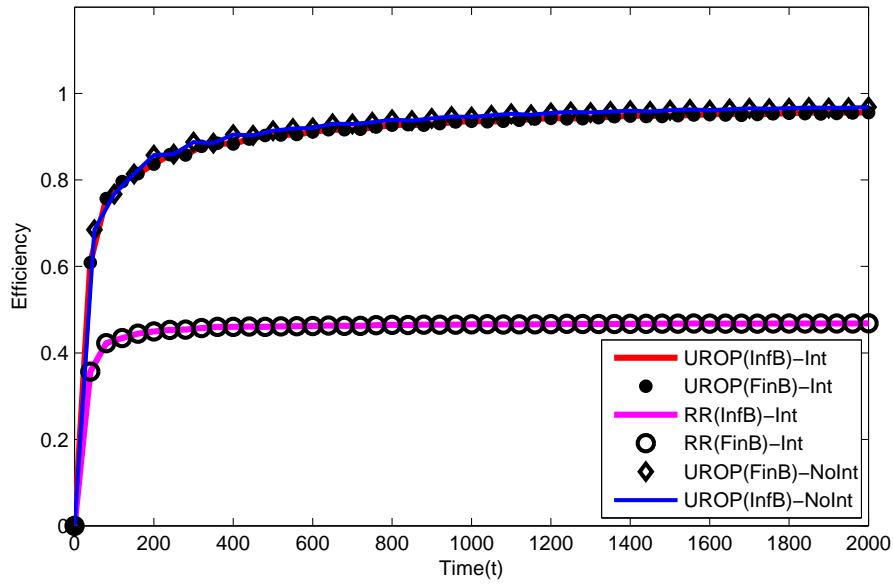


Fig. 6. Efficiencies (ratio of total throughput by a policy to total throughput by optimal policy) of UROP, RR under infinite and finite battery  $B_i=50$  assumptions for independent high density energy arrivals ( $D = 0.975$ ) such that  $m/k \in \mathbb{Z}$ . Efficiency of UROP is also shown for  $m/k$  taking a noninteger value.

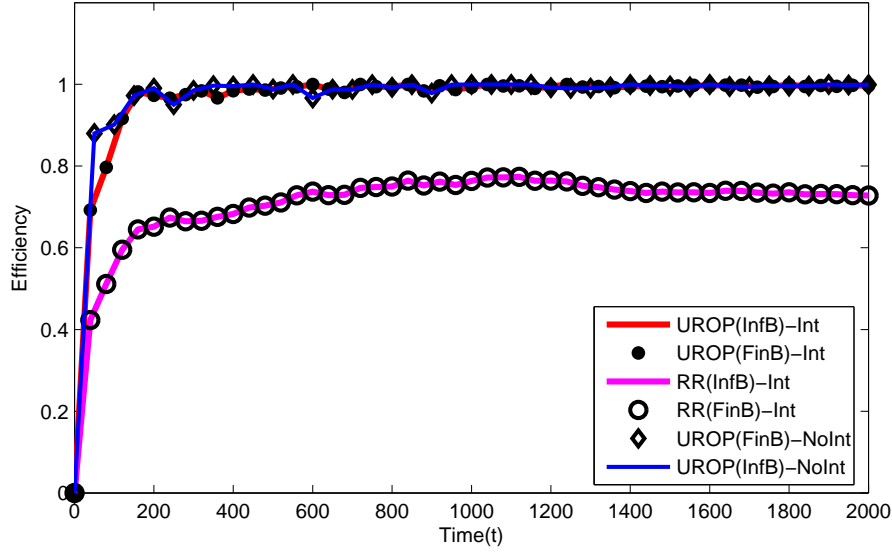


Fig. 7. Efficiencies (ratio of total throughput by a policy to total throughput by optimal policy) of UROP, RR under infinite and finite battery  $B_i=50$  assumptions for Markov low density energy arrivals ( $D = 0.2$ ) such that  $m/k \in \mathbb{Z}$ . Efficiency of UROP is also shown for  $m/k$  taking a noninteger value.

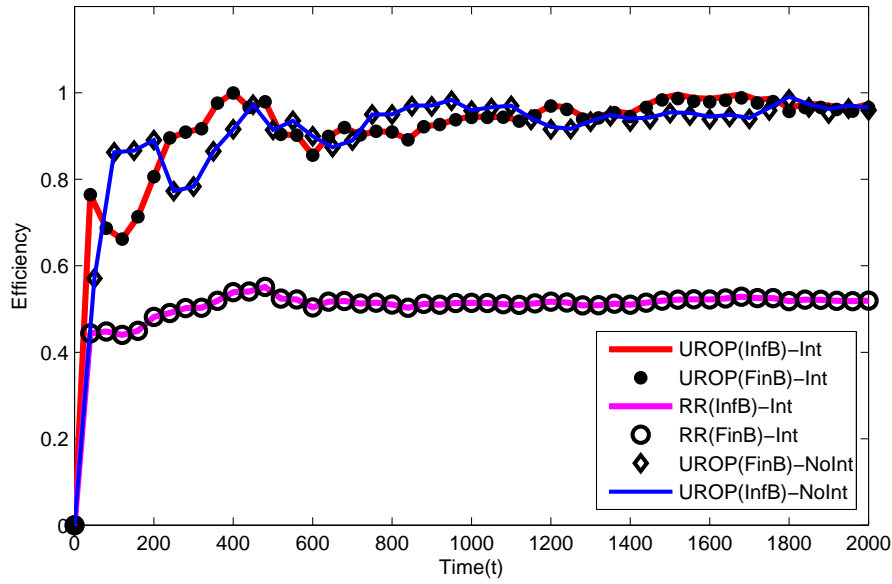


Fig. 8. Efficiencies (ratio of total throughput by a policy to total throughput by optimal policy) of UROP, RR under infinite and finite battery  $B_i=50$  assumptions for Markov low density energy arrivals ( $D = 0.975$ ) such that  $m/k \in \mathbb{Z}$ . Efficiency of UROP is also shown for  $m/k$  taking a noninteger value.

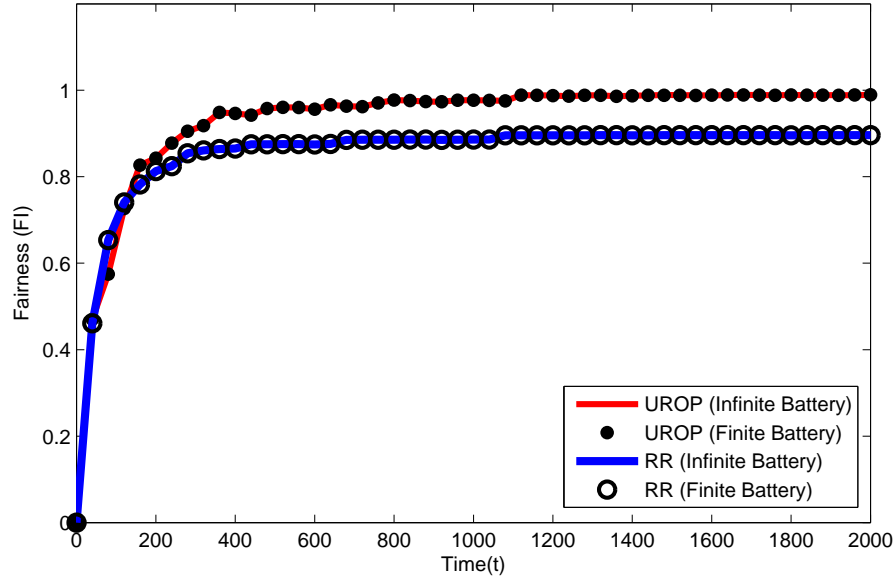


Fig. 9. Fairness of UROP, RR under infinite and finite battery  $B_i=50$  assumptions for high density  $D = 0.975$  and independent EH process by  $m/k$  integer assumption.

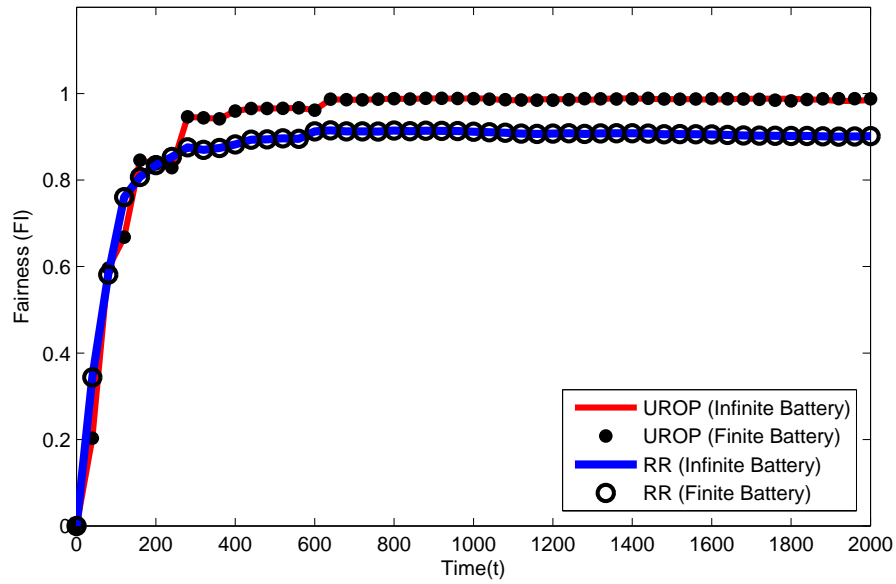


Fig. 10. Fairness of UROP, RR under infinite and finite battery  $B_i=50$  assumptions for high density  $D = 0.975$  and Markov EH process by  $m/k$  integer assumption.